SPRN059

DETAILS ON SIGNAL PROCESSING

FALL 1990 TEXAS INSTRUMENTS ISSUE 24

TMS320C40: THE FIRST PARALLEL DIGITAL SIGNAL PROCESSOR

IN THIS ISSUE...

TMS320 DSP Hotline:

713-274-2320

TI has disclosed the world's first digital signal processor designed specifically for parallel processing. The TMS320C40 parallel digital signal processing (pDSP) chip, offers designers unprecedented system performance by allowing easy connection of a virtually unlimited-number of TMS320C40s.

The 'C40 features six communication ports for direct connection between processors. An on-chip DMA coprocessor supports interprocessor communication concurrent with CPU calculations. With its high-performance 32-bit CPU and development tools designed specifically for easy development of pDSP applications, the 'C40 is an excellent building block for the performance-hungry applications of the 1990s.

Although single-chip DSPs continue to increase in performance, many designers are turning to multiple DSP solutions to obtain high performance levels. An implementation with a DSP optimized for parallel processing will yield still higher performance. The 'C40 is the highest performance 32-bit floating-point DSP available today—operating at 275 million operations per second (MOPs) and transferring data at 320 million bytes per second (Mbytes/sec)—with a 40-ns cycle time.

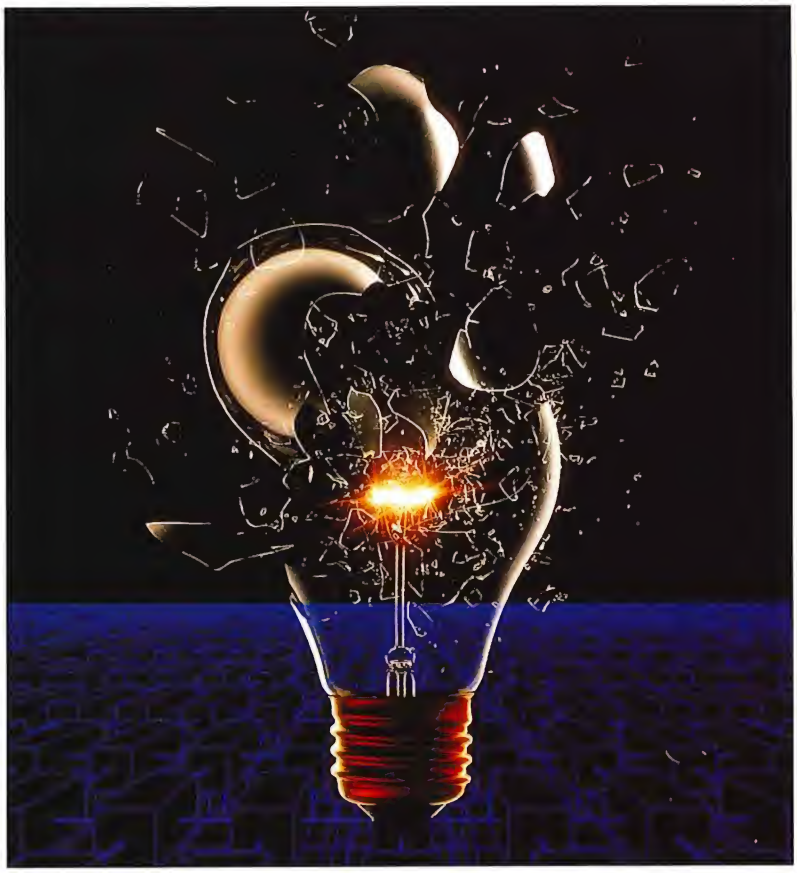

Figure 1. The TMS320C40 will shatter your perceptions about high performance

Parallel processing is being embraced to answer the needs of computationally-intensive applications such as 3D graphics, image compression, decompression and identification, neural networks, robotics, and speech recognition.

The 'C40 not only offers the industry's highest-performance *(Continued on page 2.)*

TMS320C5x named finalist for EDN Innovation of the Year

EDN Magazine has selected the TMS320C5x DSP as a finalist in the EDN Innovation of the Year Awards competition, as announced in the September 3rd edition of EDN. This issue also includes a reader survey which will determine the winner of the award.

Factors contributing to the 'C5x's innovativeness include its high level of system integration, large on-chip integration (12K of memory and 1.2 million transistors), and adoption of the IEEE JTAG test/emulation standard. Innovations like JTAG and advanced on-chip test/emulation hardware have enabled leading-edge development tools such as the 'C5x scan-based emulator and high-level language debugger, a user interface that allows trace and breakpoints while viewing running *(Continued on page 3.)*

Spectron announces OSPA for TMS320 DSPs

Spectron MicroSystems has announced the Open Signal Processing Architecture (OSPA)™ for the TI TMS320 floating-point DSPs. OSPA provides a common set of interfaces and protocols that facilitate integration of DSP hardware and software with end-user application programs executing on standard host computer systems. Based on SPOX™, the *(Continued on page 4.)*

TMS320C40: Performance tailored for your application

Does your application require more performance than you can get from a single DSP solution? The 'C40 is a single-chip building block that will allow users to get the performance they need. With the 'C40's communication ports and support for shared global memory, 'C40 processors can be connected in a variety of configurations. These ports, along with the 'C40's multichannel DMA coprocessor and high-performance 32-bit CPU, provide the ingredients to build the supersystems you have envisioned.

The 'C40 is the first true pDSP building block with communication ports to support interprocessor communications and global and local buses to support external memory in parallel systems.

A virtually unlimited variety of pDSP configurations can be achieved by connecting the pins on the communication ports without any external logic. You can construct a 2-D mesh system for image processing, or build a tree structure for speech recognition algorithms. Bidirectional rings with forward propagation are perfect for neural networks, or choose a pipelined linear array for convolution, correlation, and complex arithmetic required by radar applications.

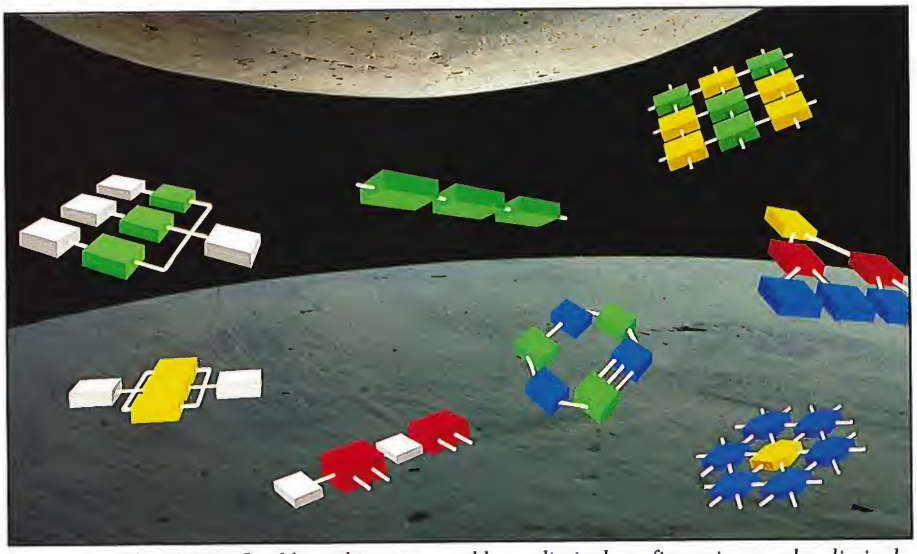

Figure 3. The 'C40's flexible architecture enables unlimited configurations and unlimited performance

The sky is the limit with the flexibility and performance of the 'C40. With TI's pDSP software and development tools, you can design a high-performance pDSP system to meet your application's performance needs.

EDN (Cont. from page 1).

assembly and C languages simultaneously. Finally, the 'C5x's modular architecture promises rapid, application-specific spin-offs in the future.

The TMS320C51 has already begun sampling and has been well-received, in telecom, computer, and control applications. Development software is available today, as is the 'C5x Software Development System (SWDS), a PC-resident board that includes a 'C51 device and allows full-speed execution of code. The 'C5x SWDS features the high-level language debugger for fast time to market. Third Party tools are also available..

To receive a copy of the TMS320C5x User's Guide (SPRU056), call the TI Customer Response Center at (800) 232-3200, x3510.

New optimization capability in 'C30 C compiler

Contributed by Alan Davis

Release 4.00 of the TMS320C30 ANSI C compiler contains advanced new optimization capabilities that can dramatically improve the performance of DSP applications written in C. The new compiler applies dozens of general optimization techniques while taking advantage of the 'C30's highly parallel architecture, such as allocating variables into registers, simplifying statements and expressions, and rearranging loops. The new optimizations often produce speed improvements of 15-25% on general purpose code and up to 1000% (10x) on loops. In addition, the new compiler reduces the memory requirements of most programs.

One of the most important considerations in compiling for a DSP is register use. The 'C30 compiler maximizes the use of registers for storage of local variables, parameters, and temporaries. By analyzing the lifetime of variables and using a weighted cost algorithm, the compiler can optimally map the set of variables into the 'C30 register set. Often, a single register can be used for several variables whose uses do not overlap.

Many DSP applications are very loop-intensive. The optimizer recognizes loops and rewrites them to execute efficiently on the 'C30. For example, a dot-product program that loops through the elements of two arrays and sums their products is written to use the 'C30's autoincrement addressing and repeat-block capabilities, so that the body of the loop is only two instructions long.

The 'C30 compiler also incorporates numerous other state-of-the-art optimizations. These include common subexpression elimination, copy propagation, inline expansion of runtime support functions, strength reduction and many more. Some of the 'C30-specific features that the compiler can use are parallel instructions, delayed branches, and zero-overhead loops. For the first time, the performance of compiled code can approach that of hand-coded assembly for even the most demanding DSP algorithms.

Release 4.00 is available now. Free updates will be shipped to registered users.

DESIGN TIPS

TMS320C30 FFT routine

Contributed by Donald G. Chandler and Nelson L. Chang, Omniplanar, Inc.

In Appendix C2 of Volume 3, Digital Signal Applications with the TMS320 Family, (pp 112-113), a C callable radix-2 real FFT program for the TMS320C30 is provided. The bit reversal section relies heavily on TI's bit-reversal addressing mode (e.g. *ARn++(IR0)B). The base address must start at an address where the last m binary digits are zero (m is the log (base 2) of the FFT size). For example, if the FFT size is 64, the base address must start at xxxxxxxxxx000000.

However, invoking the FFT subroutine with a data set from a C program leads to unpredictable results; it depends on where the linker has placed the data set in memory. One cannot easily correct this situation from C. We solved this problem with some minor modifications to the original assembly bit reversal code, as shown in Figure 5.

In this code, the lines ending with comments are modifications of the original code. We first set the two registers AR0 and AR1 to zero. This ensures that the bit addressing will work. We then define the index register IR1 to contain the address of the input data (i.e., an offset from zero). Note that we then manipulate the pointers at AR0 and AR1 on the fly; every time we need to load or store we pre-increment with the offset. These modifications do not increase the code execution time.

```
        LDI     @FFTSIZ,RC
        SUBI    1,RC
        LDI     @FFTSIZ,IR0
        LSH     -1,IR0
        SUBI    R0,R0           ;define r0 to be zero
        LDI     R0,AR0          ;set AR0 to be zero
        LDI     R0,AR1          ;set AR1 to be zero
        LDI     @INPUT,IR1      ;define IR1 to have the proper offset
        RPTB    BITRV
        CMPI    AR1,AR0
        BGE     CONT
        LDF     *+AR0(IR1),R0   ;pre-increment the index to AR0
||      LDF     *+AR1(IR1),R1   ;pre-increment the index to AR1
        STF     RO,*+AR1(IR1)   :pre-increment the index to AR1
||      STF     R1,*+AR0(IR1)   ;pre-increment the index to AR0
        CONT    NOP*ARO++
        BITRV   NOP*AR1++(IR0)B
```

*Figure 5. 'C30 bit reversal routine: fft_rl.asm**

We're Glad You Asked

Q: What do you recommend I use as decoupling capacitors for the TMS320C25?

A: We recommend that you put a large capacitor (22uF) at the point the power comes onto the board and a small capacitor (.01uF to .1uF) at each power pin on the 'C25. This applies to all TMS320 devices.

Q: What is the difference between the single-access and dual-access RAM on the TMS320C5x?

A: The single-access RAM looks like the zero wait state external RAM to the CPU—one read or write per instruction cycle. Dual-access RAM allows a read and write in a single instruction cycle.

Q: Can I configure the TMS320C50's 9K of single-access RAM to both data and program (for example, 4K data and 5K program)?

A: Yes, you can enable OVLY and RAM bit in the status register. You enable the RAM to be both data and program.

Using the new fixed-point DSP macro capability

The enhanced macro capability introduced as part of Version 6.00 of the TMS320 Fixed-Point ('C1x/C2x/C5x) DSP assembler, is based on string substitution. The macro functionality defines a "substitution symbol" which represents a character string. When the assembler encounters one of these symbols it "substitutes" the character string for the symbol. Substitution symbols can be used anywhere within a source module and to represent macro parameters.

Built-in functions enable developers to make decisions based on the string value of substitution symbols. These functions always return a value, and are most useful in logical expressions.

The example in Figure 6 is a "smart" context switch macro. The input to the macro is a list of items that should be saved by the context switch. The built-in function $ISMEMBER(ITEM, PARMS) removes the first parameter from the list and assigns it to ITEM. $SYMLEN() returns the length of a substitution symbol, and $SYMCMP() compares the string values of two character strings.

```
* CONTEXT SAVE ON SUBROUTINE CALL OR
            INTERRUPT
*
* ASSUMES AR7 IS THE STACK POINTER
*           PARMS-LIST OF ITEMS THAT ARE SAVED
*           ACC, ST, P, T
* EXAMPLE SWITCH ACC,P,T ; SAVE ACCUMULATOR,
            T, AND P REGS.
SWITCH  .MACRO  PARMS
        .VAR    ITEM
        .IF     $SYMLEN(PARMS) = 0
        .MEXIT
        .ENDIF

        LARP    AR7 ; SET UP SP
        MAR     *-
        .LOOP
        .BREAK  $ISMEMBER (ITEM, PARMS) =0
        .IF     $SYMCMP (ITEM, "ACC") = 0
        SACH    *-
        SACL    *-
        .ELSEIF $SYMCMP(ITEM, "ST") = 0
        SST1    *-
        SST     *-
        .ELSEIF $SYMCMP(ITEM, "P") = 0
        SPM     0  ; NO SHIFT ON PR OUTPUT
        SPH     *-
        SPL     *-
        .ELSEIF $SYMCMP(ITEM, "T") = 0
        MPYK    1
        SPL     *-

        .ENDIF
        .ENDLOOP
        .ENDM
```

Figure 6. Smart context switch macro